WHAT IS CLAIMED IS:

Sub A2

1. A method for predicting a result of a conditional branch instruction, comprising the steps of:

determining if a specified condition register field is used to store a branch condition of the conditional branch instruction; and

providing a software branch prediction of the conditional branch instruction as a function of the determination if the specified condition register field is used to store the branch condition of the conditional branch instruction.

2. The method as recited in claim 1, wherein the software branch prediction predicts that the conditional branch instruction will be taken if the specified condition register field is used to store the branch condition of the conditional branch instruction.

3. The method as recited in claim 2, wherein the software branch prediction predicts that the conditional branch instruction will be not taken if the specified condition register field is not used to store the branch condition of the conditional branch instruction.

Cont A2

4. The method as recited in claim 1, wherein the software branch prediction predicts that the conditional branch instruction will be not taken if the specified condition register field is used to store the branch condition of the conditional branch instruction.

5. The method as recited in claim 4, wherein the software branch prediction predicts that the conditional branch instruction will be taken if the specified condition register field is not used to store the branch condition of the conditional branch instruction.

6. The method as recited in claim 1, wherein the specified condition register field is N, where N is an integer.

7. The method as recited in claim 6, wherein the specified condition register field is a multiple of N.

Cont A2

8. A processor comprising:

an instruction fetch unit for fetching a conditional branch instruction;

circuitry for determining if a specified condition register field is used to store a branch condition of the conditional branch instruction; and

circuitry for providing a software branch prediction of the conditional branch instruction as a function of the determination if the specified condition register field is used to store the branch condition of the conditional branch instruction.

9. The processor as recited in claim 8, wherein the software branch prediction predicts that the conditional branch instruction will be taken if the specified condition register field is used to store the branch condition of the conditional branch instruction.

10. The processor as recited in claim 9, wherein the software branch prediction predicts that the conditional branch instruction will be not taken if the specified condition register field is not used to store the branch condition of the conditional branch instruction.

Cont A2

11. The processor as recited in claim 8, wherein the software branch prediction predicts that the conditional branch instruction will be not taken if the specified condition register field is used to store the branch condition of the conditional branch instruction.

12. The processor as recited in claim 11, wherein the software branch prediction predicts that the conditional branch instruction will be taken if the specified condition register field is not used to store the branch condition of the conditional branch instruction.

13. The processor as recited in claim 8, wherein the specified condition register field is N, where N is an integer.

14. The processor as recited in claim 13, wherein the specified condition register field is a multiple of N.

Sub A2 15-38 ok for 717/136

15. A method for compiling a sequence of instructions to be executed in a processor, wherein the sequence of instructions include at least one branch instruction, the method comprising the steps of:

generating the branch instruction;

determining whether to predict the branch instruction to be taken or not taken; and

storing a branch condition pertaining to the branch instruction in a condition register field specified as a function of the determined prediction.

16. The method as recited in claim 15, wherein the storing step further comprises the step of:

reordering instructions in the sequence of instructions neighboring the branch instruction so that the branch instruction is generated at a specified address.

17. The method as recited in claim 16, wherein the specified address is a multiple of a specified number N.

18. The method as recited in claim 15, wherein the storing step further comprises the step of:

generating an appropriate number of NOP instructions so that the branch instruction can be generated at a specified address.

Cont A2

19. The method as recited in claim 18, wherein the specified address is a multiple of a specified number N.

20. The method as recited in claim 15, wherein the storing step further comprises the steps of:

if the branch is predicted to be taken, determining if condition register field 4 is available;

if condition register field 4 is available, using the condition register field 4 to store the branch condition; and

generating the conditional branch instruction so that a BI field uses condition register field 4.

21. The method as recited in claim 20, wherein the storing step further comprises the steps of:

if condition register field 4 is not available, determining if condition register field 8 is available;

if condition register field 8 is available, using the condition register field 8 to store the branch condition; and

Cont A2

generating the conditional branch instruction so that the BI field uses condition register field 8.

22. The method as recited in claim 20, wherein the storing step further comprises the steps of:

if condition register field 4 is not available, generating an appropriate number of NOP instructions so that the branch instruction can be generated at a specified address.

23. The method as recited in claim 20, wherein the storing step further comprises the steps of:

if condition register field 4 is not available, reordering instructions in the sequence of instructions neighboring the branch instruction so that the branch instruction is generated at a specified address.

24. The method as recited in claim 21, wherein the storing step further comprises the steps of:

if condition register field 8 is not available, using any available condition register bit to generate a branch condition and generating the branch instruction so that it uses the same condition register field;

Cont
PR

determining if the branch instruction is at an address that is a multiple of a specified number;

if the branch instruction is at the address that is the multiple of the specified number, generating the branch instruction;

if the branch instruction is not at the address that is the multiple of the specified number, determining if the branch instruction can be reordered with neighboring instructions so that the branch instruction can be placed at an address that is the multiple of the specified number; and

if the branch instruction can be reordered with neighboring instructions so that the branch instruction can be placed at the address that is the multiple of the specified number, reordering the neighboring instructions so that the branch instruction can be placed at the address that is the multiple of the specified number.

25. The method as recited in claim 24, wherein the storing step further comprises the steps of:

if the branch instruction cannot be reordered with neighboring instructions so that the branch instruction can be placed at the address that is the multiple of the specified number, generating an appropriate number of NOP instructions so that the branch instruction can be generated at the address that is the multiple of the specified number.

Cont A2

26. The method as recited in claim 15, wherein the storing step further comprises the steps of:

if the branch is predicted to be not taken, determining if any of condition register fields 1, 2, 3, 5, 6, 7 is available;

if any of condition register fields 1, 2, 3, 5, 6, 7 is available, using one of the condition register fields 1, 2, 3, 5, 6, 7 to store the branch condition; and

generating the conditional branch instruction so that a BI field uses one of the condition register fields 1, 2, 3, 5, 6, 7.

27. The method as recited in claim 26, wherein the storing step further comprises the steps of:

determining if the branch instruction is at an address that is not a multiple of a specified number;

if the branch instruction is at the address that is not the multiple of the specified number, generating the branch instruction;

if the branch instruction is not at the address that is not the multiple of the specified number, determining if the branch instruction can be reordered with neighboring instructions so that the branch instruction can be placed at an address that is not the multiple of the specified number; and

Cont A2

if the branch instruction can be reordered with neighboring instructions so that the branch instruction can be placed at the address that is not the multiple of the specified number, reordering the neighboring instructions so that the branch instruction can be placed at the address that is not the multiple of the specified number.

28. The method as recited in claim 27, wherein the storing step further comprises the steps of:

if the branch instruction cannot be reordered with neighboring instructions so that the branch instruction can be placed at the address that is not the multiple of the specified number, generating an appropriate number of NOP instructions so that the branch instruction can be generated at the address that is not the multiple of the specified number.

Cont a2

29. A data processing system, comprising:

a processor;

a memory unit operable for storing a compiler program operable for compiling a sequence of instructions to be executed in the processor, wherein the sequence of instructions include at least one branch instruction;

an input mechanism;

an output mechanism; and

a bus system coupling the processor to the memory unit, input mechanism, and output mechanism, wherein the compiler program is operable for performing the following program steps:

generating the branch instruction;

determining whether to predict the branch instruction to be taken or not taken; and

storing a branch condition pertaining to the branch instruction in a condition register field specified as a function of the determined prediction.

30. The data processing system as recited in claim 29, wherein the storing program step further comprises the program step of:

reordering instructions in the sequence of instructions neighboring the branch instruction so that the branch instruction is generated at a specified address.

Cont A2

31. The data processing system as recited in claim 29, wherein the storing program step further comprises the program step of:

generating an appropriate number of NOP instructions so that the branch instruction can be generated at a specified address.

32. The data processing system as recited in claim 29, wherein the storing program step further comprises the program steps of:

if the branch is predicted to be taken, determining if condition register field 4 is available;

if condition register field 4 is available, using the condition register field 4 to store the branch condition; and

generating the conditional branch instruction so that a BI field uses condition register field 4.

33. The data processing system as recited in claim 32, wherein the storing program step further comprises the program steps of:

if condition register field 4 is not available, determining if condition register field 8 is available;

if condition register field 8 is available, using the condition register field 8 to store the branch condition; and

Cont A2

generating the conditional branch instruction so that the BI field uses condition register field 8.

34. The data processing system as recited in claim 33, wherein the storing program step further comprises the program steps of:

if condition register field 8 is not available, generating an appropriate number of NOP instructions so that the branch instruction can be generated at a specified address.

35. The data processing system as recited in claim 33, wherein the storing program step further comprises the program steps of:

if condition register field 8 is not available, reordering instructions in the sequence of instructions neighboring the branch instruction so that the branch instruction is generated at a specified address.

36. The data processing system as recited in claim 33, wherein the storing program step further comprises the program steps of:

if condition register field 8 is not available, using any available condition register bit to generate a branch condition and generating the branch instruction so that it uses the same condition register field;

Cont A2

determining if the branch instruction is at an address that is a multiple of a specified number;

if the branch instruction is at the address that is the multiple of the specified number, generating the branch instruction;

if the branch instruction is not at the address that is the multiple of the specified number, determining if the branch instruction can be reordered with neighboring instructions so that the branch instruction can be placed at an address that is the multiple of the specified number;

if the branch instruction can be reordered with neighboring instructions so that the branch instruction can be placed at the address that is the multiple of the specified number, reordering the neighboring instructions so that the branch instruction can be placed at the address that is the multiple of the specified number; and

if the branch instruction cannot be reordered with neighboring instructions so that the branch instruction can be placed at the address that is the multiple of the specified number, generating an appropriate number of NOP instructions so that the branch instruction can be generated at the address that is the multiple of the specified number.

Cont A2

37. The data processing system as recited in claim 29, wherein the storing program step further comprises the program steps of:

if the branch is predicted to be not taken, determining if any of condition register fields 1, 2, 3, 5, 6, 7 is available;

if any of condition register fields 1, 2, 3, 5, 6, 7 is available, using one of the condition register fields 1, 2, 3, 5, 6, 7 to store the branch condition; and

generating the conditional branch instruction so that a BI field uses the one of the condition register fields 1, 2, 3, 5, 6, 7.

38. The data processing system as recited in claim 37, wherein the storing program step further comprises the program steps of:

determining if the branch instruction is at an address that is not a multiple of a specified number;

if the branch instruction is at the address that is not the multiple of the specified number, generating the branch instruction;

if the branch instruction is not at the address that is not the multiple of the specified number, determining if the branch instruction can be reordered with neighboring instructions so that the branch instruction can be placed at an address that is not the multiple of the specified number;

Cont A2

if the branch instruction can be reordered with neighboring instructions so that the branch instruction can be placed at the address that is not the multiple of the specified number, reordering the neighboring instructions so that the branch instruction can be placed at the address that is not the multiple of the specified number; and

if the branch instruction cannot be reordered with neighboring instructions so that the branch instruction can be placed at the address that is not the multiple of the specified number, generating an appropriate number of NOP instructions so that the branch instruction can be generated at the address that is not the multiple of the specified number.

Cont A2

39. A data processing system for predicting whether a conditional branch instruction will be taken or not taken, the data processing system comprising the program steps of:

determining if the conditional branch instruction if positioned at a specified address in a sequence of instructions being executed; and

predicting whether the conditional branch instruction will be taken or not taken as a function of the position of the specified address.

40. The data processing system as recited in claim 30, wherein the predicting program step will predict taken if the specified address is a multiple of specified number N.